

Express Mail Mailing Label No.: EV260609077US
Atty. Docket No. IBX-003
(7903/6)

UNITED STATES PATENT APPLICATION

of

Wai T. Chan

Jill P. Card

and

An Cao

for

**FILTER MODELS FOR
DYNAMIC CONTROL OF COMPLEX PROCESSES**

FILTER MODELS FOR DYNAMIC CONTROL OF COMPLEX PROCESSES

CROSS-REFERENCE TO RELATED APPLICATION

[0001] This application claims priority to and the benefits of U.S. Provisional Application Serial No. 60/405,154, filed on August 22, 2003, the entire disclosure of which is hereby incorporated by reference.

FIELD OF THE INVENTION

[0002] The invention relates to the field of data processing and process control. In particular, the invention relates to the neural network control of multi-step complex processes.

BACKGROUND

[0003] The manufacture of semiconductor devices requires hundreds of processing steps. In turn, each process step may employ several process tools. Each process tool may have several manipulable parameters -- e.g. temperature, pressure and chemical concentrations -- that affect the outcome of a process step. In addition, there may be associated with each process tool several maintenance parameters that impact process performance, such as the age of replaceable parts and the time since process tool calibration.

[0004] Both process manipulable parameters and maintenance parameters associated with a process may be used as inputs for a model of the process. However, these two classes of parameters have important differences. Manipulable parameters typically exert a predictable effect and do not exhibit non-linear time-dependent behavior. Maintenance

parameters, on the other hand, affect the process outcome in a more sophisticated way. For example, the time elapsed since a maintenance event typically has a highly non-linear effect. However, the degree of non-linearity is often unknown. It is a challenge to build an accurate model of the effect of maintenance events on process outcome because prior knowledge of the degree of non-linearity is typically required for the model to be accurate. One way to handle this unknown non-linearity is to provide multiple initial estimates of the non-linear behavior for each maintenance parameter as a pre-processing step of the modeling effort, and rely on the model's ability to use only those estimates that capture the non-linear characteristics in the model. In a process model based on that approach, each maintenance parameter is represented by multiple input variables: there are typically one or more initial estimates of the non-linear behavior for each maintenance parameter.

[0005] Unfortunately, the processing time for a model typically increases exponentially with the number of input variables. The processing time may also increase as a result of inaccurate initial estimates. This approach, therefore, runs counter to the desirability of modeling complex processes with a minimum number of input variables. Accordingly, models of complex processes that avoid adding extra input variables to address the unknown behavior of other input variables, and methods for building such models, are needed.

SUMMARY OF THE INVENTION

[0006] The present invention facilitates construction of non-linear regression models of complex processes in which the outcome of the process is better predicted by the output of a function of an input variable having at least one unknown parameter that

characterizes the function than by the input variable itself. The present invention avoids the creation of extra variables in the initial input variable set and may improve the performance of model training. No initial estimates of the unknown parameter(s) that characterize the function of the input variables and related preprocesses are required. Preferably, the non-linear regression models used in the present invention comprise a neural network.

[0007] In one aspect, the present invention comprises a method of modeling a complex process having a plurality of input variables, a portion of which have unknown behavior that can be described by a function. The function, in turn, comprises at least one unknown parameter and produces an output that is a better predictor of outcome of the process than the associated input variable itself. The method comprises providing a non-linear regression model of the process and using the model to predict the outcome of the process. The model comprises a plurality of first connection weights that relate the plurality of input variables to a plurality of process metrics. The model also comprises a function and a plurality of second connection weights that relate input variables in the portion to the plurality of process metrics. Each of the plurality of second connection weights correspond to an unknown parameter associated with an input variable in the portion. In some embodiments, the plurality of second connection weights are derived by a method of building the model of a complex process. In some embodiments, the non-linear regression model has at least a first hidden layer and a last hidden layer. The first hidden layer has a plurality of nodes, each of which corresponds to an input variable with unknown behavior. In these embodiments, each node in the first hidden layer relates an input variable with the function and a second connection weight. In such embodiments,

more hidden layers may be added if the function comprises two or more unknown parameters.

[0008] In another aspect, the present invention comprises a method of building a non-linear regression model of a complex process having a plurality of input variables. A portion of the input variables exhibit unknown behavior that can be described by a function having at least one unknown parameter. These input variables may, in some embodiments, be input variables for a first hidden layer of the model having a plurality of nodes. In these embodiments, each node in the first hidden layer is associated with one of the input variables and has a single synaptic weight. In accordance with the method, a function of an input variable that has at least one unknown parameter and whose output is a predictor of output of the process is identified. A model comprising a plurality of connection weights that relate the plurality of input variables to a plurality of process metrics is provided, and an error signal for the model is determined. The one or more unknown parameters of the function and the plurality of connection weights are adjusted in a single process based on the error signal. In some embodiments, the one or more unknown parameters initially comprise values that are randomly assigned. In other embodiments, the one or more unknown parameters initially comprise the same arbitrarily assigned value. In other embodiments, the one or more unknown parameters initially comprise one or more estimated values. For example, the error signal may be used in part to determine a gradient for a plurality of outputs of the first hidden layer, and the adjustment may be made to one or more of the synaptic weights corresponding to one or more unknown parameters of the function. The adjustment process (e.g., to one or more of the synaptic weights) is repeated until a convergence criterion is satisfied.

[0009] In some embodiments, the invention involves the model of a complex process that features a set of initial input variables comprising both manipulated variables and maintenance variables. As used herein, the term “manipulable variables” refers to input variables associated with the manipulable parameters of a process. The term “manipulable variables” includes, for example, process step controls that can be manipulated to vary the process procedure. One example of a manipulable variable is a set point adjustment. As used herein, the term “maintenance variables” refers to input variables associated with the maintenance parameters of a process. The term “maintenance variables” includes, for example, variables that indicate the wear, repair, or replacement status of a sub-process component(s) (referred to herein as “replacement variables”), and variables that indicate the calibration status of the process controls (referred to herein as “calibration variables”).

[0010] In various embodiments, the non-linear regression model comprises a neural network. A neural network can be organized as a series of nodes (which may themselves be organized into layers) and connections among the nodes. Each connection is given a weight corresponding to its strength. For example, in one embodiment, the non-linear regression model comprises a first hidden layer that serves as a filter for specific input variables (organized as nodes of an input layer with each node corresponding to a separate input variable) and at least a second hidden layer that is connected to the first hidden layer and the other input variables (also organized as nodes of an input layer with each node corresponding to a separate input variable). The first hidden layer utilizes a single neuron (or node) for each input variable to be filtered.

[0011] The second hidden layer may be fully connected to the first hidden layer and to the input variables that are not connected to the first hidden layer. In some embodiments, the second layer is not directly connected to the input variables that are connected to the first hidden layer, whereas in other embodiments, the second hidden layer is fully connected to the first hidden layer and to all of the input variables.

[0012] In one embodiment, the outputs of the second hidden layer are connected to the outputs of the non-linear regression model, i.e., the output layer. In other embodiments, the non-linear regression model comprises one or more hidden layers in addition to the first and second hidden layers; accordingly, in these embodiments the outputs of the second hidden layer are connected to another hidden layer instead of the output layer.

[0013] In some embodiments, the function associated with an input variable comprises two unknown parameters. In some such embodiments, the non-linear regression model comprises two hidden filter layers having a plurality of nodes each corresponding to an input variable in the portion. Such embodiments involve filtering the input variables with the two hidden filter layers, using a synaptic weight for each input variable and each hidden filter layer. Each of these synaptic weights corresponds to one of the two unknown parameters in the function.

[0014] In other aspects, the present invention provides systems adapted to practice the aspects of the invention set forth above. In some embodiments of these aspects, the present invention provides an article of manufacture in which the functionality of portions of one or more of the foregoing methods of the present invention are embedded on a computer-readable medium, such as, but not limited to, a floppy disk, a hard disk, an optical disk, a magnetic tape, a PROM, an EPROM, CD-ROM, or DVD-ROM.

[0015] In another aspect, the invention comprises an article of manufacture for building a non-linear regression model of a complex process having a plurality of input variables, a portion of which have unknown behavior that can be described by a function comprising at least one unknown parameter. The function produces an output that is a predictor of the outcome of the process. The article of manufacture includes a process monitor, a memory device, and a data processing device. The data processing device is in signal communication with the process monitor and the memory device. The process monitor provides data representing the plurality of input variables and the corresponding plurality of process metrics. The memory device provides the function and a plurality of first weights corresponding to the at least one unknown parameter associated with each of input variables in the portion. In some embodiments, the plurality of second connection weights comprise values that are randomly assigned. In other embodiments, the plurality of second connection weights all comprise the same arbitrarily assigned initial value. In other embodiments, the plurality of second connection weights comprise one or more estimated values. The data processing device receives the data, the function, and the plurality of first weights and determines an error signal of the model from them. The data processing device adjusts the plurality of first weights and a plurality of second weights that relate a plurality of input variable to the plurality of process metrics, in a single process based on the error signal.

[0016] In embodiments of the foregoing aspect, the data processing device determines the error signal for the output layer of the model and uses the error signal to determine a gradient for the output of the function associated with each input variable in the portion, and adjust the weight corresponding to the at least one unknown parameter accordingly.

[0017] In embodiments of the foregoing aspect, the data processing device also determines if a convergence criterion is satisfied. In some such embodiments, the data processing device will adjust the weights again if the convergence criterion is not satisfied or terminate the process if the convergence criterion is satisfied.

[0018] In another aspect, the invention comprises an article of manufacture for modeling a complex process having a plurality of input variables, a portion of which have unknown behavior that can be described by a function comprising at least one unknown parameter. The function produces an output that is a predictor of the outcome of the process. The article of manufacture includes a process monitor, a memory device, and a data processing device. The data processing device is in signal communication with the process monitor and the memory device. The process monitor provides data representing the plurality of input variables. The memory device provides a plurality of first connection weights that relate the plurality of input variables to a plurality of process metrics, the function, and a plurality of second weights corresponding to the at least one unknown parameter associated with each of input variables in the portion. In some embodiments, the plurality of second weights are derived by an article of manufacture for building a non-linear regression model of a complex process. The data processing device receives the plurality of input variables, the plurality of first connection weights, the function, and the plurality of second connection weights; and predicts an outcome of the complex process in a single process using that information.

[0019] In embodiments of the foregoing aspects, the process monitor comprises a database or a memory element including a plurality of data files. In some embodiments, the data representing input variables and process metrics include binary values and scalar

numbers. In some such embodiments, one or more of scalar numbers is normalized with a zero mean. In embodiments of the foregoing aspects, the memory device is any device capable of storing information, such as a floppy disk, a hard disk, an optical disk, a magnetic tape, a PROM, an EPROM, CD-ROM, or DVD-ROM. In some such embodiments, the memory device stores information in digital form. In embodiments of the foregoing aspects, the memory device is part of the process monitor. In embodiments of the foregoing aspects, the data processing device comprises a module embedded on a computer-readable medium, such as, but not limited to, a floppy disk, a hard disk, an optical disk, a magnetic tape, a PROM, an EPROM, CD-ROM, or DVD-ROM.

[0020] In various embodiments of the foregoing aspects, the function for the unknown behavior is non-linear with respect to the input variable. In some such embodiments, the input variable represents a time elapsed since an event associated with the complex process. In one such embodiment, the function is of the form $\exp(-\lambda_j y_j)$ where λ_j is the synaptic weight associated with an input y_j , and wherein the input y_j is an input variable of the portion of the plurality input variables. The input y_j in such an embodiment may represent the time elapsed since a maintenance event. In various embodiments, the input variables comprise, but are not limited to, continuous values, discrete values, and binary values.

[0001] In some embodiments of the foregoing aspects, the adjustment is of the form $\Delta\lambda_j = -\eta y_j \delta_j$ where η is a learning rate parameter, δ_j is the gradient of an output of a node j of the first hidden layer with the input y_j , $\Delta\lambda_j$ is the adjustment for synaptic weight λ_j associated with the input y_j , and the input y_j is an input variable of the portion of the plurality input variables.

BRIEF DESCRIPTION OF THE DRAWINGS

[0022] A more complete understanding of the advantages, nature, and objects of the invention may be attained from the following illustrative description and the accompanying drawings. The drawings are not necessarily drawn to scale, and like reference numerals refer to the same parts throughout the different views.

[0023] Figure 1A is a schematic representation of one embodiment of a non-linear regression model for a complex process according to the present invention;

[0024] Figure 1B is a schematic representation of another embodiment of a non-linear regression model for a complex process according to the present invention;

[0025] Figure 1C is a schematic representation of a third embodiment of a non-linear regression model for a complex process according to the present invention;

[0026] Figures 2 is a flow diagram illustrating building a non-linear regression model according to one embodiment of the present invention; and

[0027] Figures 3A and 3B are a flow diagram illustrating one embodiment of building a non-linear regression model according to the present invention.

[0028] Figure 4 is a system in accordance with embodiments of the present invention.

ILLUSTRATIVE DESCRIPTION

[0029] An illustrative description of the invention in the context of a neural network model of a complex process follows. However, one of ordinary skill in the art will understand that the present invention may be used in connection with other non-linear regression models that have input variables with unknown behavior and that describe

complex processes whose outcome is better predicted by a function of such variables than by the input variables themselves.

[0030] In the illustrative example, the initial non-linear regression model comprises a neural network model. As illustrated in Figures 1A, 1B, and 1C, the neural network model 100 has $m + n$ input variables y . The first m input variables (y_1, \dots, y_m) 102 are variables to be filtered. In some embodiments, these m variables represent maintenance variables, which have an unknown non-linear, time-dependent behavior that affects process outcome. The remaining n input variables $(y_{m+1}, \dots, y_{m+n})$ 104 are variables that will not be filtered. In this example, these n variables represent manipulated variables that do not exhibit non-linear time behavior. The first hidden layer 105 of the neural network comprises m nodes 107 (indexed by j) and serves as a filter layer for the maintenance variables 102. There is a one-to-one connection between the input nodes 1 through m and the filter layer nodes 107. If we denote the nodes in this first layer 105 by node 1 through m , then for $j=1, \dots, m$, the input to node j is y_j with a synaptic weight λ_j . Thus, no extra input variables are added to model the maintenance variables.

[0031] In the embodiments illustrated in FIG. 1A and 1B, each node 107 in the first hidden layer 105 has an activation function with one unknown parameter. In the illustrative embodiment in particular, the activation function associated with each node 107 in the first hidden layer 105 is an exponential function of the form:

$$\varphi(x) = \exp(-x) \quad \text{Eq. (1).}$$

This choice of exponential function is related to a practice in reliability engineering, which models the reliability of a part at age t by the exponential distribution $\exp(-\lambda t)$.

As a result, the output from the first hidden layer 105 for each node j is $\exp(-\lambda_j y_j)$.

[0032] In one alternative embodiment, the activation function is another parametric form of the reliability function. In other embodiments, the activation function comprises, for example, a Weibull distribution, $\exp(-\lambda_j y_j^{\beta_j})$, a lognormal distribution, and a gamma distribution, $\int_0^x (x^{\alpha-1} e^{-x}) / \Gamma(\alpha) dx$. These are the typical probability models used in engineering and biomedical applications. Accordingly, it is to be understood that the present invention is not limited to exponential activation functions.

[0033] Referring to Figure 1A, in one embodiment, the second hidden layer 109, contains K nodes 111 where each node $k = 1, \dots, K$ is connected to each node 107 of the first hidden layer 105 in accordance with the respective connection weight (i.e., the nodes are fully connected) and is also connected to each of the input manipulated variables 104. The second hidden layer 109 is in turn fully connected to the output layer 114 (i.e., all nodes 111 can contribute to the value of each of the nodes 113 in the output layer).

[0034] Referring to the alternative illustrative embodiment of Figure 1B, there is again a one-to-one connection between the input nodes 1 through m and the nodes of the first hidden layer 105. Unlike in the embodiment of Figure 1A, the K nodes 111 in the second hidden layer 109 are directly connected to each of the input maintenance variables 102 as well as to each node 107 of the first hidden layer 105 and to each of the input manipulated variables 104. Thus, if the maintenance variables 102 have other contributions that are not sufficiently captured by the first hidden layer 105, the model can compensate by adjusting the weights directly from the input maintenance nodes (variables) 102. As in Figure 1A, the second hidden layer 109 is also fully connected to the output layer 114.

[0035] In an embodiment that incorporates an activation function with two unknown parameters, a non-linear regression model such as that illustrated in FIG. 1C may be used. As in Figures 1A and 1B, the model depicted in Figure 1C features a one-to-one connection between the input nodes 1 through m and the nodes of the first hidden layer 105. Unlike in the embodiments of Figures 1A and 1B, however, Figure 1C features a second hidden filter layer 120 between the first hidden layer 105 and hidden layer 109. There is a one-to-one connection between the nodes of the first hidden layer 105 and the nodes of hidden filter layer 120. In some embodiments there is also a one-to-one connection between the input layer 102 and the nodes of hidden filter layer 120. Thus, there is one filter layer associated with each unknown parameter in the filter function. The k nodes 111 in hidden layer 109 are connected to each node j of hidden layer 120 and to each of the input manipulated variables 104. As in Figures 1A and 1B, hidden layer 109 is also fully connected to the output layer 114 in Figure 1C.

[0036] As in the embodiments of Figures 1A and 1B, each node 107 in the first hidden layer 105 of Figure 1C has an activation function with one unknown parameter. In the embodiment illustrated in Figure 1C, each node in hidden layer 120 also has an activation function with one unknown parameter. As an illustrative example, the Weibull distribution can be implement using Figure 1C as follows: If the input to node j in layer 102 is y_j , an input of $\log(y_j)$ will be fed forward to a node in layer 105. The synaptic weight between a node in layer 102 and layer 105 may be designated β_j and the synaptic weight between a node in layer 105 and layer 120 may be designated λ_j . Each node in hidden layer 105 has activation function of the form $\varphi(x) = \exp(x)$ and each node in hidden layer 120 has activation function of the form $\varphi(x) = \exp(-x)$. As a result, the

output from the first hidden layer 105 for each node j is $\exp(\beta_j \log(y_j)) = y_j^{\beta_j}$ and the output from the second hidden layer 120 for each node j is $\exp(-\lambda_j y_j^{\beta_j})$. Thus, no extra input variables are added to model the maintenance variables.

[0037] In an alternative embodiment similar to Figure 1B, the K nodes 111 in Figure 1C are also directly connected to each of the input maintenance variables 102 to capture any contributions that are not sufficiently captured by hidden layers 105 and 120.

[0038] The present invention also provides methods and systems for building non-linear regression models that incorporate such a filter layer. The model building begins with the recognition that one or more input variables are not optimally used to predict output of the process directly. Instead, the input variable is a better predictor of the output of the process after it has been pre-processed or filtered. In particular, there is a function of the input variable whose output is a better predictor of the output of the process than the input variable itself. This function, however, is characterized by at least one unknown parameter and therefore cannot be used directly. The function may be referred to as an activation function. The filter layer enables at least one unknown parameter in the function to be estimated and the output of the function to be used as the predictor of the output of the process.

[0039] The non-linear regression model of the illustrative example is built by comparing a calculated output variable, based on measured maintenance and manipulated variables for an actual process run, with a target value based on the actual output variables as measured for the actual process run. The difference between calculated and target values (such as, e.g., measured process metrics), or the error, is used to compute the corrections to the adjustable parameters in the regression model. Where the regression model is a

neural network as in the illustrative example, these adjustable parameters are the connection weights between the nodes in the network.

[0040] Figure 2 illustrates the basic process of building a non-linear regression model of a complex process that incorporates a filter layer in accordance with the invention. In step 210, an activation function of an input variable is identified. The output of the function is a predictor of the outcome of the complex process. The function, however, is characterized by at least one unknown parameter. The function is typically identified based on knowledge about the relationship between an input variable and the outcome of the process.

[0041] In step 220, an error signal for an output layer of the non-linear regression model in accordance with the embodiments is determined. In step 230, a gradient for each of the outputs of the first hidden layer is determined using the error signal. In step 240, an adjustment to one or more of the synaptic weights corresponding to one or more unknown parameters is determined. In the model itself and in the process of building the model, only those synaptic weights between the input layer and the one or more filter layers correspond to one or more unknown parameters of an activation function. Other synaptic weights in the model may be calculated, for example, using standard equations known to be useful for calculating such weights in neural networks. An embodiment of the invention featuring steps similar to step 220 through step 240 is described in detail below with respect to Figures 3A and 3B.

[0042] In optional step 250 of Figure 2, a convergence criterion is evaluated. If the convergence criterion is not satisfied, steps 210 through 250 are repeated. In one embodiment, the process is repeated using the same set of input variables and

corresponding output variables measured from an actual run of the process. In another embodiment, the process is repeated using a different set of input variables and corresponding output variables measured from an actual run of the process. If the convergence criterion is satisfied, the process ends and the model is complete.

[0043] Illustrated in Figures 3A and 3B is a flow diagram of one embodiment of a process for building a non-linear regression model, in this example a neural network, having $p + 1$ layers L_v (where $v = 0, 1, \dots, p-1, p$), inclusive of an input layer $L_{v=0}$ and an output layer $L_{v=p}$. As used in Figures 3A and 3B, the indices i, j, k and layer designations I, J and K have the following meanings: the index i spans the nodes of a layer I; the index j spans the nodes of a layer J; and the index k spans the nodes of a layer K, where the output of layer I serves as the input to layer J and the output of layer J serves as the input to layer K.

[0044] Referring to Figure 3A, the building approach starts with the output layer $J = L_p$ and its predecessor layer $I = L_{p-1}$ (block 305) to determine the output layer error signals e_j (block 310); accordingly, no layer K is used at this stage. As illustrated in Figure 3A, the output layer L_p error signals e_j may be determined from

$$e_j = d_j - z_j \quad \text{Eq. (2),}$$

where d_j represents the desired output (or target value) of node j and z_j represents the actual output value of node j . The error signals e_j are then used to adjust the weights w_{ji} connecting layers I and J (block 315). The adjustment Δw_{ji} to a weight w_{ji} may be determined from

$$\Delta w_{ji} = \eta \delta_j z_i \quad \text{Eq. (3),}$$

where η denotes the learning-rate parameter, δ_j is the gradient of error against node inputs x_j for the output of node j , and z_i represents the output of node I (i.e., the input through connection weight w_{ji} in to node j). The gradient δ_j may be determined from

$$\delta_j = f'_j(x_j) e_j \quad \text{Eq. (4),}$$

where f_j is the activation function for node j .

[0001] After the weights w_{ji} are adjusted to $(w_{ji} + \Delta w_{ji})$, the approach is continued back through the non-linear regression model. In accordance with Figures 3A and 3B, now layer $I = L_{a=p-2}$, layer $J = L_{b=p-1}$ and layer $K = L_{c=p}$ (blocks 317, 320, and 325). As a result, the weights w_{kj} connecting layers J and K are the previously determined adjusted weights $(w_{ji} + \Delta w_{ji})$ (block 315).

[0046] The approach back-propagates through the non-linear regression model using the gradient δ_k at the output of the nodes k to determine the error signals e_j of the new layer $J=L_b$ (block 330). For example, at a node j the gradient δ_j is the product of $f'_j(x_j)$ and the weighted sum of the δ s computed for the nodes in layer K that are connected to node j . Accordingly, the layer J error signals e_j may be determined from,

$$e_j = \sum_k w_{kj} \delta_k \quad \text{Eq. (5),}$$

and the gradient δ_j from,

$$\delta_j = f'_j(x_j) \sum_k w_{kj} \delta_k \quad \text{Eq. (6),}$$

where the summing of both equations (5) and (6) occurs over all nodes in layer K that are connected to layer J. The error signals e_j are then used to adjust the weights w_{ji}

connecting layers I and J (block 340). This adjustment Δw_{ji} to a weight w_{ji} may then be determined from

$$\Delta w_{ji} = \eta z_i f_j'(x_j) \sum_k w_{kj} \delta_k \quad \text{Eq. (7),}$$

as illustrated in Figure 3B.

[0001] The approach continues to back-propagate the error signals layer by layer through the non-linear regression model until the gradients δ_j of the nodes j of the first hidden layer $J = L_1$ can be determined (i.e., until $I = L_{a=0}$ and the answer to query 350 is “YES”). As previously discussed, the activation function $f(x)$ used in the illustrative embodiment for the filtered input variables is of the form $\varphi(x) = \exp(-x)$, and the inputs to a node are y_j and λ_j where y_j is the j th input to the neural network and λ_j is the synaptic weight of connection between the j th node in the input layer and the j th node in the first hidden layer. The gradient δ_j at node j may then be given by

$$\delta_j = -\exp(-\lambda_j y_j) \sum_{k \in C_j} w_{kj} \delta_k \quad \text{Eq. (8),}$$

where C_j is the set of nodes in the second hidden layer K that are connected to node j .

[0048] The building approach then adjusts the synaptic weights λ_j of the activation function (block 360) using the gradients δ_j . Thus, the adjustment $\Delta \lambda_j$ to the synaptic weight λ_j may be given by

$$\Delta \lambda_j = -\eta y_j \delta_j = -\eta y_j \exp(-\lambda_j y_j) \sum_{k \in C_j} w_{kj} \delta_k \quad \text{Eq. (9).}$$

[0049] The building approach of Figures 3A and 3B is then repeated until the change in the adjustment terms $\Delta\lambda_j$ satisfies a convergence criterion. A typical convergence criterion first defines a tolerance factor which indicates a meaningful improvement in the average prediction accuracy over all training records. If the convergence criterion is satisfied (“YES” to query 370) then the building round is ended. If the convergence criterion is not satisfied (“NO” to query 370) then the outputs of the model, i.e., the values of the nodes of the output layer L_p , are recalculated (block 380) using the adjusted connection weights ($w_{ji} + \Delta w_{ji}$) and adjusted synaptic weights ($\lambda_{ji} + \Delta\lambda_{ji}$). The process of error signal determination and weight correction is then repeated (action 390). The process is thus preferably repeated until the convergence criterion is satisfied. In one such embodiment, the process is not repeated if the average prediction accuracy has not improved within the tolerance factor for a pre-determined number of process iterations.

[0050] The building approach illustrated by Figures 3A and 3B may be utilized with a single set of target values d_j (e.g., a set of measured maintenance and manipulated variables and measured output values for a single process run, or a set of averaged measured maintenance and manipulated variables and measured output values for a plurality of process runs) or multiple sets of target values d_j .

[0001] Preferably, the building approach of the present invention is conducted for a plurality of sets of target values d_j . For example, in one embodiment, the building approach conducts a first building run utilizing a first set of target values d_j and determines synaptic weight adjustments until a first convergence criterion is satisfied. The approach then uses the adjusted connection weights ($w_{ji} + \Delta w_{ji}$) and adjusted synaptic weights ($\lambda_{ji} + \Delta\lambda_{ji}$) determined in the first building run to conduct a second building run

utilizing a second set of target values d_j and determines synaptic weight adjustments until a second convergence criterion is satisfied. The approach continues with additional building runs utilizing third, fourth, etc., sets of target values d_j with the adjusted weights from the prior building run.

[0052] In other aspects, the present invention provides systems and articles of manufacture adapted to practice the methods of the invention set forth above. In embodiments illustrated by Figure 4, the system comprises a process monitor 410, a memory device, and a data processing device 430. In these embodiments, the data processing device 430 is in signal communication with the process monitor 410 and the memory device 420. A system or article of manufacture in accordance with Figure 4 may build a non-linear regression model of a complex process having a plurality of input variables, a portion of which exhibit unknown behavior that can be described by a function comprising at least one unknown parameter, or model such a process, or both.

[0053] The process monitor 410 may comprise any device that provides data representing input variables and/or corresponding process metrics associated with the process. The process monitor 410 in some embodiments, for example, comprises a database that includes data from process sensor, yield analyzers, or the like. In related embodiments, the process monitor 410 is a set of files from a statistical process control database. Each file in the process monitor 410 may represent information relating to a specific process. The information may include binary values and scalar numbers. The binary values may indicate relevant technology and equipment used in the process. The scalar numbers may represent process metrics. The process metrics may be normalized. The normalization may have a zero mean and/or a unity standard deviation.

[0054] The memory device 420 illustrated in Figure 4 may comprise any device capable of storing a function, a plurality of first weights representing at least one unknown parameter from the function associated with an input variable in the portion, and, in some embodiments, a plurality of second weights that relate the plurality of input variables to the plurality of process metrics. In some embodiments, the plurality of weights initially comprise values that are randomly assigned. In other embodiments, the plurality of weights initially comprise the same arbitrarily assigned initial value. In other embodiments, the plurality of weights initially comprise one or more estimated values. The memory device 420 provides the stored information to the data processing device 430. A memory device 420 may, for example, be a floppy disk, a hard disk, an optical disk, a magnetic tape, a PROM, an EPROM, CD-ROM, or DVD-ROM. In some such embodiments, the memory device stores information in digital form. The memory device 420 in some embodiments, for example, comprises a database. The memory device 420 in some embodiments is part of the process monitor 410. In some embodiments, the invention further comprises a user interface that enables the function and/or weights in the memory device 420 to be input or directly modified by the user.

[0055] The data processing device 430 may comprise an analog and/or digital circuit adapted to implement portions of the functionality of one or more of the methods of the present invention using at least in part data from the process monitor 410 and the function from the memory device 420. In some embodiments, the data processing device 430 uses data from the process monitor 410 to adjust the weights in the memory device 420. In some embodiments, the data processing device 430 sends the adjusted weights back to the memory device 420 for storage. In some such embodiments, the data processing

device 430 may adjust a weight by determining the error signal for the output layer of the model and using the error signal to determine a gradient for the output of the function. In some such embodiments, the data processing device 430 also evaluates a convergence criterion and adjusts the weights again if the criterion is not met. In other embodiments, the data processing device 430 uses the function and the weights in the memory device 420, along with input variable from the process monitor 410, to predict outcome of the process. In addition, in one embodiment, data processing device 430 is adapted to adjust the weights after a process outcome is predicted thereby improving the model and its filtering continually.

[0056] In some embodiments, the data processing device 430 may implement the functionality of portions of the methods of the present invention as software on a general-purpose computer. In addition, such a program may set aside portions of a computer's random access memory to provide control logic that affects the non-linear regression model implementation, non-linear regression model training and/or the operations with and on the input variables. In such an embodiment, the program may be written in any one of a number of high-level languages, such as FORTRAN, PASCAL, C, C++, Tcl, or BASIC. Further, the program can be written in a script, macro, or functionality embedded in commercially available software, such as EXCEL or VISUAL BASIC. Additionally, the software could be implemented in an assembly language directed to a microprocessor resident on a computer. For example, the software can be implemented in Intel 80x86 assembly language if it is configured to run on an IBM PC or PC clone. The software may be embedded on an article of manufacture including, but not limited to,

“computer-readable program means” such as a floppy disk, a hard disk, an optical disk, a magnetic tape, a PROM, an EPROM, or CD-ROM.